VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI – 590018, Karnataka

INTERNSHIP REPORT

ON

A PREDICTIVE MODEL FOR FORECASTING DEMAND AND SUPPLY INFORMATION OF TOP CROPS

BACHELOR OF ENGINEERING IN
COMPUTER SCIENCE AND
ENGINEERING

Submitted by:

SAKETH N SHET
4JN21CS136

Conducted at

COMPSOFT TECHNOLOGIES

J.N.N COLLEGE OF ENGINEERINGSHIVAMOGGA

DEPARTMENTOF COMPUTER SCIENCE AND ENGINEERING

Accredited by NBA, New Delhi
NAVULE,SHIMOGA

J.N.N COLLEGE OF ENGINEERINGSHIVAMOGGA

DEPARTMENTOF COMPUTER SCIENCE AND ENGINEERING

Accredited by NBA, New Delhi
NAVULE,SHIMOGA.

CERTIFICATE

This is to certify that the Internship titled **"**A PREDICTIVE MODEL FOR FORECASTING DEMAND AND SUPPLY INFORMATION OF TOP CROPS**"** carried out by **Mr SAKETH N SHET,** a bonafide student of JNN COLLEGE OF ENGINEERING, in partial fulfillment for the award of **Bachelor of Engineering**, in **COMPUTER SCIENCE AND ENGINEERING BRANCH** under Visvesvaraya Technological University, Belagavi, during the year 2022-2023. It is certified that all corrections/suggestions indicated have been incorporated in the report.

The project report has been approved as it satisfies the academic requirements in respect of Internship prescribed for the course Internship / Professional Practice (21CSI85)

Signature of Guide **Signature of HOD** **Signature of Principal**

External Viva:

Name of the Examiner Signature with Date

1)_____
_____

2)_____
_____

D E C L A R A T I O N

I, SAKETH N SHET,third year student of computer science and engineering,JAWAHARLAL NEHRU NEW COLLEGE OF ENGINEERING, declare that the Internship has been successfully completed, in **COMPSOFT TECHNOLOGIES**. This report is submitted in partial fulfillment of the requirements for award of Bachelor Degree in Branch name, during the academic year 2022-2023.

Date :05-12-2023 :

Place:SHIMOGA

USN : 4JN21CS136

NAME : SAKETH N SHET

OFFER LETTER

Internship Offer Letter

Date: 25th October, 2023

Name: **Saketh N Shet**
USN: **4JN21CS136**
Placement ID: **23OCTMLBONE**

Dear Student,

We would like to congratulate you on being selected for the **Machine Learning with Python (Research Based)** Internship position with **Compsoft Technologies**, effective Start Date **25th October, 2023**, All of us are excited about this opportunity provided to you!

This internship is viewed as being an educational opportunity for you, rather than a part-time job. As such, your internship will include training/orientation and focus primarily on learning and developing new skills and gaining a deeper understanding of concepts of **Machine Learning with Python (Research Based)** through hands-on application of the knowledge you learn while you train with the senior developers. You will be bound to follow the rules and regulations of the company during your internship duration.

Again, congratulations and we look forward to working with you!.

Sincerely,

Nithin K. S
Project Manager
Compsoft Technologies
No. 363, 19th main road,
1st Block Rajajinagar
Bangalore - 560010

A C K N O W L E D G E M E N T

This Internship is a result of accumulated guidance, direction and support of several important persons. We take this opportunity to express our gratitude to all who have helped us to complete the Internship.

We express our sincere thanks to our Principal, for providing us adequate facilities to undertake this Internship.

We would like to thank our Head of Dept – branch code, for providing us an opportunity to carry out Internship and for his valuable guidance and support.

We would like to thank our Lab assistant Software Services for guiding us during the period of internship.

We express our deep and profound gratitude to our guide, Guide name, Assistant/Associate Prof, for her keen interest and encouragement at every step in completing the Internship.

We would like to thank all the faculty members of our department for the support extended during the course of Internship.

We would like to thank the non-teaching members of our dept, forhelping us during the Internship.

Last but not the least, we would like to thank our parents and friends without whose constant help, the completion of Internship would have not been possible.

USN : 4JN21CS136

NAME : SAKETH N SHET

ABSTRACT

It is evident that there is a huge gap between demand and supply of various crops, due to which both farmers and consumers are facing problems. At present, in India there is no system in place to efficiently manage this demand and supply issue. The potential of present-day technologies like data analytics, machine learning can be exploited to overcome these issues.

The available data about the demand, supply, price variation of the crops and other factors affecting the supply chain of agricultural produce can be used to analyse and come up with a model to predict and forecast market variations of agricultural crops.

The supply is less for any commodities, as consumers have to compete with one other to buy the less supplied goods, results in increased price for the commodity, making consumers suffer with the high price.

As there is no synchronization in production and demand for the agricultural commodities, either farmer fail to get good market prices for their products, or consumer suffers high prices due to less production.

Table of Contents

1. COMPANY PROFILE

A Brief History of Compsoft Technologies

Compsoft Technologies, was incorporated with a goal ”To provide high quality and optimal Technological Solutions to business requirements of our clients”. Every business is a different and has a unique business model and so are the technological requirements. They understand this and hence the solutions provided to these requirements are different as well. They focus on clients requirements and provide them with tailor made technological solutions. They also understand that Reach of their Product to its targeted market or the automation of the existing process into e-client and simple process are the key features that our clients desire from Technological Solution they are looking for and these are the features that we focus on while designing the solutions for their clients.

Sarvamoola Software Services. is a Technology Organization providing solutions for all web design and development, MYSQL, PYTHON Programming, HTML, CSS, ASP.NET and LINQ. Meeting the ever increasing automation requirements, Sarvamoola Software Services. specialize in ERP, Connectivity, SEO Services, Conference Management, effective web promotion and tailor-made software products, designing solutions best suiting clients requirements.

Compsoft Technologies, strive to be the front runner in creativity and innovation in software development through their well-researched expertise and establish it as an out of the box software development company in Bangalore, India. As a software development company, they translate this software development expertise into value for their customers through their professional solutions.

They understand that the best desired output can be achieved only by understanding the clients demand better. Compsoft Technologies work with their clients and help them to defiine their exact solution requirement. Sometimes even they wonder that they have completely redefined their solution or new application requirement during the brainstorming session, and here they position themselves as an IT solutions consulting group comprising of high caliber consultants.

They believe that Technology when used properly can help any business to scale and achieve new heights of success. It helps Improve its efficiency, profitability, reliability; to put it in one sentence ” Technology helps you to Delight your Customers” and that is what we want to achieve.

2. ABOUT THE COMPANY

Compsoft Technologies is a Technology Organization providing solutions for all web design and development, MYSQL, PYTHON Programming, HTML, CSS, ASP.NET and LINQ. Meeting the ever increasing automation requirements, Compsoft Technologies specialize in ERP, Connectivity, SEO Services, Conference Management, effective web promotion and tailor-made software products, designing solutions best suiting clients requirements. The organization where they have a right mix of professionals as a stakeholders to help us serve our clients with best of our capability and with at par industry standards. They have young, enthusiastic, passionate and creative Professionals to develop technological innovations in the field of Mobile technologies, Web applications as well as Business and Enterprise solution. Motto of our organization is to "Collaborate with our clients to provide them with best Technological solution hence creating Good Present and Better Future for our client which will bring a cascading a positive effect in their business shape as well". Providing a Complete suite of technical solutions is not just our tag line, it is Our Vision for Our Clients and for Us, We strive hard to achieve it.

Products of Compsoft Technologies.

Android Apps

It is the process by which new applications are created for devices running the Android operating system. Applications are usually developed in Java (and/or Kotlin; or other such option) programming language using the Android software development kit (SDK), but other development environments are also available, some such as Kotlin support the exact same Android APIs (and bytecode), while others such as Go have restricted API access.

The Android software development kit includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator based on QEMU, documentation, sample code, and zutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.5.8 or later, and Windows 7 or later. As of March 2015, the SDK is not available on Android itself, but softwaredevelopment is possible by using specialized Android applications.

Web Application

It is a client–server computer program in which the client (including the user interface and client- side logic) runs in a web browser. Common web applications include web mail, online

retail sales, online auctions, wikis, instant messaging services and many other functions. web applications use web documents written in a standard format such as HTML and JavaScript,which are supported by a variety of web browsers. Web applications can be considered as a specifific variant of client–server software where the client software is downloaded to the client machine when visiting the relevant web page, using standard procedures such as HTTP. The Client web software updates may happen each time the web page is visited. During the session, the web browser interprets and displays the pages, and acts as the universal client for any web application. The use of web application frameworks can often reduce the number of errors in a program, both by making the code simpler, and by allowing one team to concentrate on the framework while another focuses on a specifified use case. In applications which are exposed to constant hacking attempts on the Internet, security-related problems can be caused by errors in the program.

Frameworks can also promote the use of best practices such as GET after POST. There are some who view a web application as a two-tier architecture. This can be a “smart” client that performs all the work and queries a “dumb” server, or a “dumb” client that relies on a “smart” server. The client would handle the presentation tier, the server would have the database (storage tier), and the business logic (application tier) would be on one of them or on both. While this increases the scalability of the applications and separates the display and the database, it still doesn‟t allow for true specialization of layers, so most applications will outgrow this model. An emerging strategy for application software companies is to provide web access to software previously distributed as local applications. Depending on the type of application, it may require the development of an entirely different browser-based interface, or merely adapting an existing application to use different presentation technology. These programs allow the user to pay a monthly or yearly fee for use of a software application without having to install it on a local hard drive. A company which follows this strategy is known as an application service provider (ASP), and ASPs are currently receiving much attention in the software industry.

Security breaches on these kinds of applications are a major concern because it can involve both enterprise information and private customer data. Protecting these assets is an important part of any web application and there are some key operational areas that must be included in the development process. This includes processes for authentication, authorization, asset handling, input, and logging and auditing. Building security into the applications from the beginning can be more effective and less disruptive in the long run.

Web design

It is encompasses many different skills and disciplines in the production and maintenance of websites. The different areas of web design include web graphic design; interface design;

authoring, including standardized code and proprietary software; user experience design; an search engine optimization. The term web design is normally used to describe the design process relating to the front-end (client side) design of a website including writing mark up. Web design partially overlaps web engineering in the broader scope of web development. Web designers are expected to have an awareness of usability and if their role involves creating mark up then they are also expected to be up to date with web accessibility guidelines. Web design partially overlaps web engineering in the broader scope of web development.

Departments and services offered

Compsoft Technologies plays an essential role as an institute, the level of education, development of student's skills are based on their trainers. If you do not have a good mentor then you may lag in many things from others and that is why we at Compsoft Technologies gives you the facility of skilled employees so that you do not feel unsecured about the academics. Personality development and academic status are some of those things which lie on mentor's hands. If you are trained well then you can do well in your future and knowing its importance of Compsoft Technologies always tries to give you the best.

They have a great team of skilled mentors who are always ready to direct their trainees in the best possible way they can and to ensure the skills of mentors we held many skill development programs as well so that each and every mentor can develop their own skills with the demands of the companies so that they can prepare a complete packaged trainee.

Services provided by Compsoft Technologies.

- Core Java and Advanced Java
- Web services and development
- Dot Net Framework
- Python
- Selenium Testing
- Conference / Event Management Service
- Academic Project Guidance
- On The Job Training
- Software Training

3. INTRODUCTION

Introduction to ML

Machine learning is a subfield of artificial intelligence that involves the development of algorithms and statistical models that enable computers to improve their performance in tasks through experience. These algorithms and models are designed to learn from data and make predictions or decisions without explicit instructions. There are several types of machine learning, including supervised learning, unsupervised learning, and reinforcement learning. Supervised learning involves training a model on labeled data, while unsupervised learning involves training a model on unlabeled data. Reinforcement learning involves training a model through trial and error. Machine learning is used in a wide variety of applications, including image and speech recognition, natural language processing, and recommended model systems.

Problem Statement

“A predictive model for forecasting demand and supply information of TOP crops”

Built a python application that analyses the top crops at any given time, depending on the season or demand. You can use the dataset available on the Internet to use.

4. SYSTEM ANALYSIS

Existing System:

The necessary data required for this analysis has been gathered from the sources like Ministryof-Agriculture, Agmarknet, Directorate of marketing and Inspection, Ministry-of- Agriculture and Farmers-Welfare, Government of India, National Horticulture Board (NHB) India, HOPCOMS Horticultural Producer's Cooperative Marketing and Processing Society Limited by applying Web scraping methods and stored in the local repository and detailed market survey.
Data preprocessing module removes the noise from the collected data sets and builds the missing values before applying forecasting algorithm for better performance. All the data sets collected were integrated into a single dataset. During this process, incomplete information is eliminated and all NA (Not Applicable) values are aggregated to the average value. The data sets collected contains multiple attributes, the required attributes are separated and stored as a separate data frame, and then the data frame is converted to time series data. The pre-processed data has been stored and processed as clusters in distributed mode for effective application of the algorithm and analysis. The Hadoop HDFS and Map Reduce paradigm have been used to provide a distributed data storing and parallel .

Proposed System:

The proposed DPFM model is implemented using the integrated R-Hadoop machine learning based prediction modeling that provides scalable and parallel processing environment. Also, a Map-Reduce programming model has been developed in R environment to perform the efficient analysis of the data stored in Hadoop clusters. The server that runs R submits the jobs to Hadoop which in turn distributes the work among m machines in the cluster and gets the result.

5. REQUIREMENT ANALYSIS

Hardware Requirement Specification

The software requirement specification can produce at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by established a complete information description, a detailed functional description, a representation of system behavior, and indication of performance and design constrain, appropriate validate criteria, and other information pertinent to requirements.

Hardware Requirement:

- System : i3 2.4 GHz & above
- Hard Disk : 256GB
- Ram : 4GB

Software Requirements:

- Operating system : Windows 10
- Coding Language : Python
- Application : Jupyter Notebook

. There are a lot of python libraries which could be used to build visualization like matplotlib, vispy, bokeh, seaborn, pygal, folium, plotly. Of the many, matplotlib and seaborn seems to be very widely used for basic to intermediate level of visualizations.

However, two of the above are widely used for visualization i.e.

- Matplotlib: It is an amazing visualization library in Python for 2D plots of arrays, It is a multiplatform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. Use the below command to install this library:

```
pip install matplotlib
```

Seaborn: This library sits on top of matplotlib. In a sense, it has some flavors of matplotlib while from the visualization point, its is much better than matplotlib and has added features as well. Use the below command to install this library:

```
pip install seaborn
```

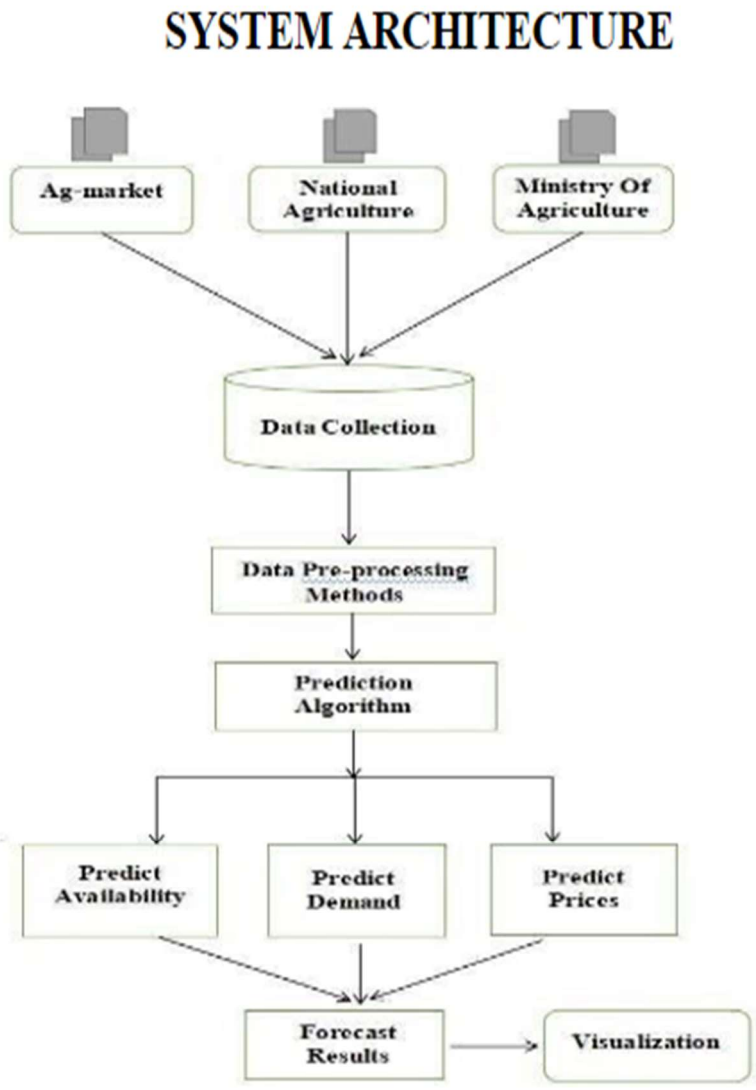
Step-by-step Approach

- Import required modules Load the dataset.
- Display the data and constraints of the loaded dataset.

Use different methods to visualize various illustrations from the data

6.DESIGN ANALYSIS

This consists of modules for data collection, data repository, data pre-processing, clustering, MapReduce and forecasting the demand.



System Architecture

7.IMPLEMENTATION

Implementation is the stage where the theoretical design is turned into a working system. The most crucial stage in achieving a new successful system and in giving confidence on the new system for the users that it will work efficiently and effectively.

The system can be implemented only after thorough testing is done and if it is found to work according to the specification. It involves careful planning, investigation of the current system and it constraints on implementation, design of methods to achieve the change over and an evaluation of change over methods a part from planning.

Two major tasks of preparing the implementation are education and training of the users and testing of the system. The more complex the system being implemented, the more involved will be the system analysis and design effort required just for implementation.

The implementation phase comprises of several activities. The required hardware and software acquisition is carried out. The system may require some software to be developed. For this, programs are written and tested. The user then changes over to his new fully tested system and the old system is discontinued.

TESTING

The testing phase is an important part of software development. It is the Information zed system will help in automate process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied. Software testing is carried out in three steps:

1. The first includes unit testing, where in each module is tested to provide its correctness, validity and also determine any missing operations and to verify whether the objectives have been met. Errors are noted down and corrected immediately.
2. Unit testing is the important and major part of the project. So errors are rectified easily in particular module and program clarity is increased. In this project entire system is divided into several modules and is developed individually. So unit testing is conducted to individual modules.
3. The second step includes Integration testing. It need not be the case, the software whose modules when run individually and showing perfect results, will also show perfect results when run as a whole.

8.CODE

```
import numpy as np
import pandas as pd
import os
for dirname, _, filenames in os.walk('/Agriculture Crop/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))


# %%
import numpy as np
import pandas as pd
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
import plotly as py
import os


# %%
df=pd.read_csv('./CROP.csv')

# %%
df.head()

# %%
df.tail()

# %%
#Load the data using pandas read functions
d1 = pd.read_csv("./CROP1.csv")
d2 = pd.read_csv("./CROP.csv")
d3 = pd.read_csv("./CROP2.csv")
d4 = pd.read_csv("./CROP3.csv")

# %%
d1 = pd.read_csv("./datafile1.csv")
d1 = d1.rename(columns={
    'Crop': 'Crop',
```

```
    'State': 'State',
    'Cost of Cultivation (`/Hectare) A2+FL': 'Cost_A2_FL',
    'Cost of Cultivation (`/Hectare) C2': 'Cost_C2',
    'Cost of Production (`/Quintal) C2': 'Cost_Production',
    'Yield (Quintal/ Hectare) ': 'Yield'
})

d1.head().style.set_properties(**{'background-
color':'lightblue','color':'black','border-color':'#8b8c8c'})

# %%
#check the shape of the data
print(f' The dataset contains {d1.shape[1]} columns and {d1.shape[0]}
rows')
# Print the column names
print(d1.columns)

# %%
d1.info()
d1.describe()

# %%
d1.isnull().sum().sum()

# %%
#checking the null values in the data
d1.isna().sum()/len(d1)*100

# %%
d1.fillna(0, inplace=True)

# %%
#Lets check the duplicate values in the data
print('The duplicate values in the data is', d1.duplicated().sum())

# %%
# Find the Average yield of the top 10 yield in the data
# Group by 'Yield' and calculate the mean for specific columns
avg_yield = d1.groupby('Yield')[['Cost_A2_FL', 'Cost_C2',
'Cost_Production']].mean()

# Display the top 10 rows with a heatmap-style background
avg_yield.head(10).style.background_gradient(cmap='Pastel1')
```

```
# %%
def state1(row):
    if 'Andhra Pradesh' in row['Recommended Zone']:
        return 1
def state2(row):
    if 'Tamil Nadu' in row['Recommended Zone']:
        return 1
def state3(row):
    if 'Gujarat' in row['Recommended Zone']:
        return 1
def state4(row):
    if 'Orissa' in row['Recommended Zone']:
        return 1
def state5(row):
    if 'Punjab' in row['Recommended Zone']:
        return 1
def state6(row):
    if 'Haryana' in row['Recommended Zone']:
        return 1
def state7(row):
    if 'Uttar Pradesh' in row['Recommended Zone']:
        return 1
def state8(row):
    if 'Rajasthan' in row['Recommended Zone']:
        return 1
def state9(row):
    if 'Karnataka' in row['Recommended Zone']:
        return 1
def state10(row):
    if 'Madhya Pradesh' in row['Recommended Zone']:
        return 1
def state11(row):
    if 'West Bengal' in row['Recommended Zone']:
        return 1



# %%
cols = d1.columns
d1.groupby('Crop')[cols[:-1]].sum().plot(kind='bar', figsize=(12,6)),
plt.title('Cost of Cultivation vs. Cost of Production by Crop')

# %%
import plotly.graph_objects as go
```

```
import pandas as pd

# Extract unique crop names from the dataset
crops = d2['Crop'].unique()

# Limit the number of crops to display (e.g., the first six crops)
crops_to_display = crops[:6]

# Create traces for each crop to display
traces = []
for crop in crops_to_display:
    trace = go.Scatter(
        x=d2.columns[1:],
        y=d2.loc[d2['Crop'] == crop, d2.columns[1:]].values.flatten(),
        mode='lines+markers',
        name=crop,
    )
    traces.append(trace)

# Create the layout
layout = go.Layout(
    title='Crop Growth Over Time',
    xaxis=dict(title='Year'),
    yaxis=dict(title='Percentage'),
)

# Create the figure
fig = go.Figure(data=traces, layout=layout)

# Show the figure
fig.show()

# %%
import matplotlib.pyplot as plt
x=[20,6,8,10]
y=[1,5,900,11]
# plt.plot(x,y, marker='*', markersize=10, linestyle='dashed',
linewidth=1, color='black')
plt.subplot(2,2,1)
plt.plot(x,y, '^:g')
plt.xlabel('Cost ')
plt.ylabel('Cost V2')
plt.title('agriculture-crop')
plt.legend(['Cost of Production by Crop', 'B'])
```

```
plt.grid()

plt.subplot(2,2,2)
plt.plot(x,y, '^-g')
plt.xlabel('Cost ')
plt.ylabel('Cost V2')
plt.title('agriculture-crop')
plt.legend(['Cost of Production by Crop', 'B'])
plt.grid()

plt.subplot(2,2,3)
plt.bar(x, y, width=0.2, color='r')
plt.xlabel('Cost ')
plt.ylabel('Cost V2')
plt.title('agriculture-crop')
plt.legend(['Cost of Production by Crop', 'B'])
plt.grid()


plt.subplot(2,2,4)


plt.show()


# %%
x = [2,2,3,50,3,6]
print(x)
plt.hist(x, color='r')

plt.xlabel('Cost')
plt.ylabel('Cost v2 ')
plt.title('agriculture crop')
plt.legend(['Cost of Production by Crop', 'B'])
plt.grid()
plt.show()

#
plt.pie(d1["State"].value_counts(),autopct="%.f%%",labels=d1["State"].u
nique())
plt.show()

# %%
import pandas as pd
```

```
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

# %%
data = pd.read_csv("./CROP1.csv")

# %%
data.head()

# %%
crop_production_data = pd.read_csv("./CROP2.csv")

# %%
crop_production_data.head()

# %%
fig,axs = plt.subplots(figsize=(10,6))
crop_wise_yield = data.groupby(['Crop']).sum()['Yield (Quintal/
Hectare) ']
plt.plot(crop_wise_yield)
crop_wise_production = data.groupby(['Crop']).sum()['Cost of Production
(`/Quintal) C2']/10
plt.plot(crop_wise_production)
plt.xticks(rotation ='vertical')
plt.legend()

# %%
state_crop_yield = data.groupby(['State'])
index = list(state_crop_yield.indices.keys())
state_crop_yield.sum()[['Cost of Production (`/Quintal) C2', 'Yield
(Quintal/ Hectare) ']].plot(kind='bar',figsize=(12,7))

# %%
recommended_zone = pd.read_csv('./CROP3.csv')

# %%
recommended_zone.drop('Unnamed: 4',axis=1,inplace=True)
recommended_zone.dropna(inplace=True)

# %%
recommended_zone.info()

# %%
```

```
recommended_zone.head()

# %%
def state1(row):
    if 'Andhra Pradesh' in row['Recommended Zone']:
        return 1
def state2(row):
    if 'Tamil Nadu' in row['Recommended Zone']:
        return 1
def state3(row):
    if 'Gujarat' in row['Recommended Zone']:
        return 1
def state4(row):
    if 'Orissa' in row['Recommended Zone']:
        return 1
def state5(row):
    if 'Punjab' in row['Recommended Zone']:
        return 1
def state6(row):
    if 'Haryana' in row['Recommended Zone']:
        return 1
def state7(row):
    if 'Uttar Pradesh' in row['Recommended Zone']:
        return 1
def state8(row):
    if 'Rajasthan' in row['Recommended Zone']:
        return 1
def state9(row):
    if 'Karnataka' in row['Recommended Zone']:
        return 1
def state10(row):
    if 'Madhya Pradesh' in row['Recommended Zone']:
        return 1
def state11(row):
    if 'West Bengal' in row['Recommended Zone']:
        return 1

# %%
recommended_zone['Andhra Pradesh'] =
recommended_zone.apply(state1,axis=1)
recommended_zone['Tamil Nadu']=recommended_zone.apply(state2,axis=1)
recommended_zone['Gujarat']=recommended_zone.apply(state3,axis=1)
recommended_zone['Orissa']=recommended_zone.apply(state4,axis=1)
recommended_zone['Punjab']=recommended_zone.apply(state5,axis=1)
```

```
recommended_zone['Haryana']=recommended_zone.apply(state6,axis=1)
recommended_zone['Uttar Pradesh']=recommended_zone.apply(state7,axis=1)
recommended_zone['Rajasthan']=recommended_zone.apply(state8,axis=1)
recommended_zone['Karnataka']=recommended_zone.apply(state9,axis=1)
recommended_zone['Madhya
Pradesh']=recommended_zone.apply(state10,axis=1)
recommended_zone['West Bangal']=recommended_zone.apply(state11,axis=1)

# %%
recommended_zone.fillna(0).head()

# %%
dataframe =
recommended_zone.groupby('Crop').sum().plot(kind='bar',figsize=(15,7))
dataframe

# %%
crop_production_data.columns = ['Crop', 'Production 2006-07',
'Production 2007-08',
       'Production 2008-09', 'Production 2009-10', 'Production 2010-
11',
       'Area 2006-07', 'Area 2007-08', 'Area 2008-09', 'Area 2009-10',
       'Area 2010-11', 'Yield 2006-07', 'Yield 2007-08', 'Yield 2008-
09',
       'Yield 2009-10', 'Yield 2010-11']

# %%
plt.subplots(figsize=(15,6))
plt.scatter(x='Crop',y='Production 2006-07',data =
crop_production_data)
plt.xticks(rotation=90)
plt.show()
```

FRONT END:

#HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Crop Analyzer</title>

    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 20px;
        }

        h1 {
            text-align: center;
            color: #333;
        }

        table {
            width: 100%;
            border-collapse: collapse;
            margin-top: 20px;
        }

        th, td {
            border: 1px solid #ddd;
            padding: 12px;
            text-align: left;
        }

        th {
            background-color: #f2f2f2;
        }

        form {
            margin-top: 20px;
            text-align: center;
        }

        label {
            margin-right: 10px;
        }

        button {
            padding: 10px 20px;
            background-color: #3ecd43;
            color: rgb(246, 238, 238);
            border: none;
            border-radius: 4px;
            cursor: pointer;
```

```
        }

        button:hover {
            background-color: #45a049;
        }
    </style>
</head>
<body>
<body background="FARM.jpg"></body>
    <h1>Crop Analyzer</h1>

    <form id="cropForm">
        <label for="season">Select Season:</label>
        <select id="season" name="season">
            <option value="">All Seasons</option>
            <option value="Winter">Winter</option>
            <option value="Spring">Spring</option>
            <option value="Summer">Summer</option>
            <option value="Fall">Fall</option>
        </select>

        <label for="demand">Demand Threshold:</label>
        <input type="number" id="demand" name="demand" placeholder="Enter
demand threshold">

        <button type="button" onclick="analyzeCrops()">Analyze Crops</button>
    </form>

    <table id="cropTable">
        <tr>
            <th>Crop</th>
            <th>Season</th>
            <th>Demand Matric ton</th>
        </tr>
        <!-- Table rows will be populated dynamically using JavaScript -->
    </table>

    <script>
        function analyzeCrops() {
            var season = document.getElementById('season').value;
            var demand = document.getElementById('demand').value;

            // Perform AJAX request or use mock data to fetch and display
results
            // For simplicity, using mock data here
            var mockData = [
                { Crop: 'Wheat', Season: 'Winter', Demand: 500 },
                { Crop: 'Rice', Season: 'Summer', Demand: 700 },
                { Crop: 'Maize', Season: 'Summer', Demand: 600 },
                { Crop: 'Barley', Season: 'Spring', Demand: 400 },
                { Crop: 'Potato', Season: 'Fall', Demand: 300 },
                { Crop: 'Tomato', Season: 'Summer', Demand: 450 },
                { Crop: 'Legume', Season: 'Winter', Demand: 500 },
                { Crop: 'Chili pepper', Season: 'Summer', Demand: 700 },
                { Crop: 'Groundnut', Season: 'Summer', Demand: 600 },
```

```
                { Crop: 'Pearl millet', Season: 'Spring', Demand: 400 },
                { Crop: 'Sugarcane', Season: 'Fall', Demand: 300 },
                { Crop: 'Brinjal', Season: 'Summer', Demand: 450 },
                { Crop: 'Millets', Season: 'Spring', Demand: 400 },
                { Crop: 'Mung bean', Season: 'Fall', Demand: 300 },
                { Crop: 'Soybean', Season: 'Summer', Demand: 450 }
            ];

            // Filter data based on user input
            var filteredData = mockData.filter(function (crop) {
                return (!season || crop.Season === season) && (!demand ||
crop.Demand >= demand);
            });

            // Populate the table with the filtered data
            var tableBody =
document.getElementById('cropTable').getElementsByTagName('tbody')[0];
            tableBody.innerHTML = ''; // Clear existing rows

            filteredData.forEach(function (crop) {
                var row = tableBody.insertRow();
                var cell1 = row.insertCell(0);
                var cell2 = row.insertCell(1);
                var cell3 = row.insertCell(2);

                cell1.innerHTML = crop.Crop;
                cell2.innerHTML = crop.Season;
                cell3.innerHTML = crop.Demand;
            });
        }
    </script>

</body>
</html>
```

9.SNAPSHOTS

[1]:
```
import numpy as np
import pandas as pd
import os
for dirname, _, filenames in os.walk('/Agriculture Crop/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
```

[2]:
```
import numpy as np
import pandas as pd
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import plotly as py
import os
```

[3]:
```
df=pd.read_csv('./Datafile.csv')
```

[4]:
```
df.head()
```

[4]:

	Crop	2004-05	2005-06	2006-07	2007-08	2008-09	2009-10	2010-11	2011-12
0	Rice	100.0	101.0	99.0	105.0	112.0	121.0	117.0	110.0
1	Wheat	100.0	101.0	112.0	115.0	117.0	127.0	120.0	108.0
2	Coarse Cereals	100.0	107.0	110.0	115.0	113.0	123.0	122.0	136.0
3	Pulses	100.0	108.0	134.0	124.0	124.0	146.0	137.0	129.0
4	Vegetables	100.0	109.0	103.0	118.0	113.0	124.0	128.0	115.0

[6]:
```
#Load the data using pandas read functions
d1 = pd.read_csv("./datafile1.csv")
d2 = pd.read_csv("./datafile.csv")
d3 = pd.read_csv("./datafile (2).csv")
d4 = pd.read_csv("./datafile (3).csv")
```

[7]:
```
d1 = pd.read_csv("./datafile1.csv")
d1 = d1.rename(columns={
    'Crop': 'Crop',
    'State': 'State',
    'Cost of Cultivation (`/Hectare) A2+FL': 'Cost_A2_FL',
    'Cost of Cultivation (`/Hectare) C2': 'Cost_C2',
    'Cost of Production (`/Quintal) C2': 'Cost_Production',
    'Yield (Quintal/ Hectare) ': 'Yield'
})

d1.head().style.set_properties(**{'background-color':'lightblue','color':'black','border-color':'#8b8c8c'})
```

[7]:

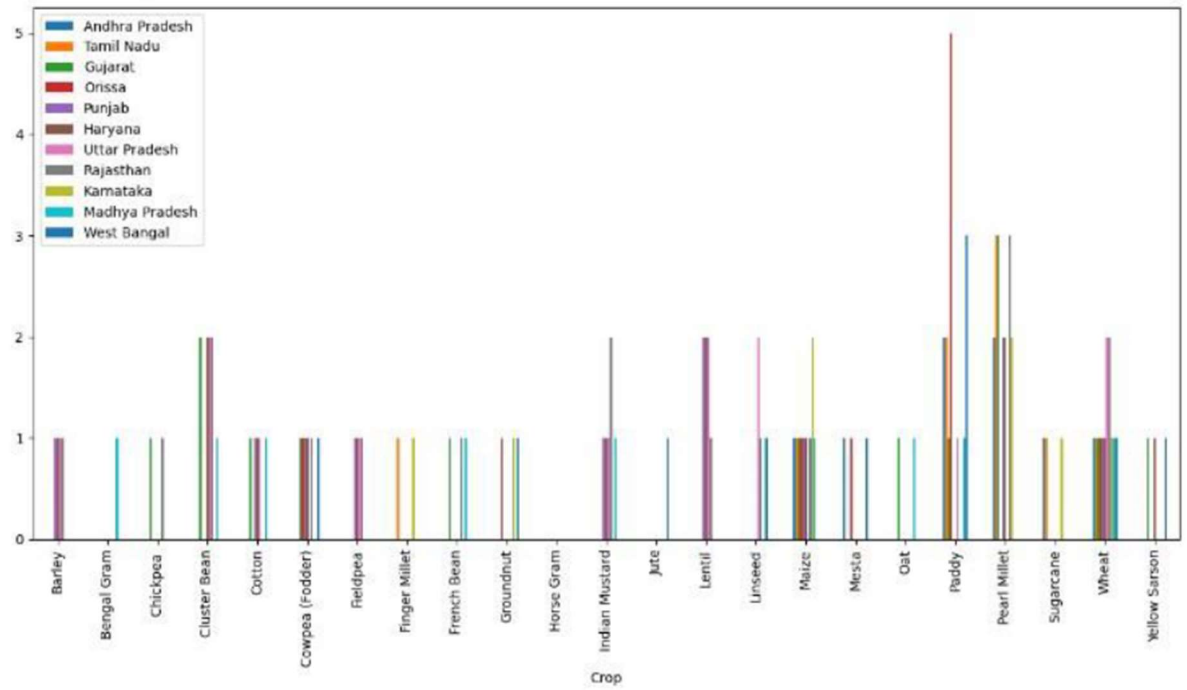
	Crop	State	Cost_A2_FL	Cost_C2	Cost_Production	Yield
0	ARHAR	Uttar Pradesh	9794.050000	23076.740000	1941.550000	9.830000
1	ARHAR	Karnataka	10593.150000	16528.680000	2172.460000	7.470000
2	ARHAR	Gujarat	13468.820000	19551.900000	1898.300000	9.590000
3	ARHAR	Andhra Pradesh	17051.660000	24171.650000	3670.540000	6.420000
4	ARHAR	Maharashtra	17130.550000	25270.260000	2775.800000	8.720000

```
# Find the Average yield of the top 10 yield in the data
# Group by 'Yield' and calculate the mean for specific columns
avg_yield = d1.groupby('Yield')[['Cost_A2_FL', 'Cost_C2', 'Cost_Production']].mean()

# Display the top 10 rows with a heatmap-style background
avg_yield.head(10).style.background_gradient(cmap='Pastel1')
```

Yield	Cost_A2_FL	Cost_C2	Cost_Production
1.320000	6440.640000	7868.640000	5777.480000
3.010000	5483.540000	8266.980000	2614.140000
4.050000	6204.230000	9165.590000	2068.670000
4.710000	13647.100000	17314.200000	3484.010000
5.900000	6684.180000	13209.320000	2228.970000
6.420000	17051.660000	24171.650000	3670.540000
6.700000	10780.760000	15371.450000	2261.240000
6.830000	8552.690000	12610.850000	1691.660000
7.470000	10593.150000	16528.680000	2172.460000
8.050000	12985.950000	18679.330000	2277.680000

```
dataframe = recommended_zone.groupby('Crop').sum().plot(kind='bar',figsize=(15,7))
dataframe
```

```
<Axes: xlabel='Crop'>
```




```
plt.figure(figsize=(12,6))
k=px.sunburst(d1,path=['State','Crop'],values='Yield',
              hover_data=['Yield'], color_continuous_scale='Blues')  # Specify the color scale here)
k.update_layout(title='Best Yield Capacity Crop')
k.show()
```

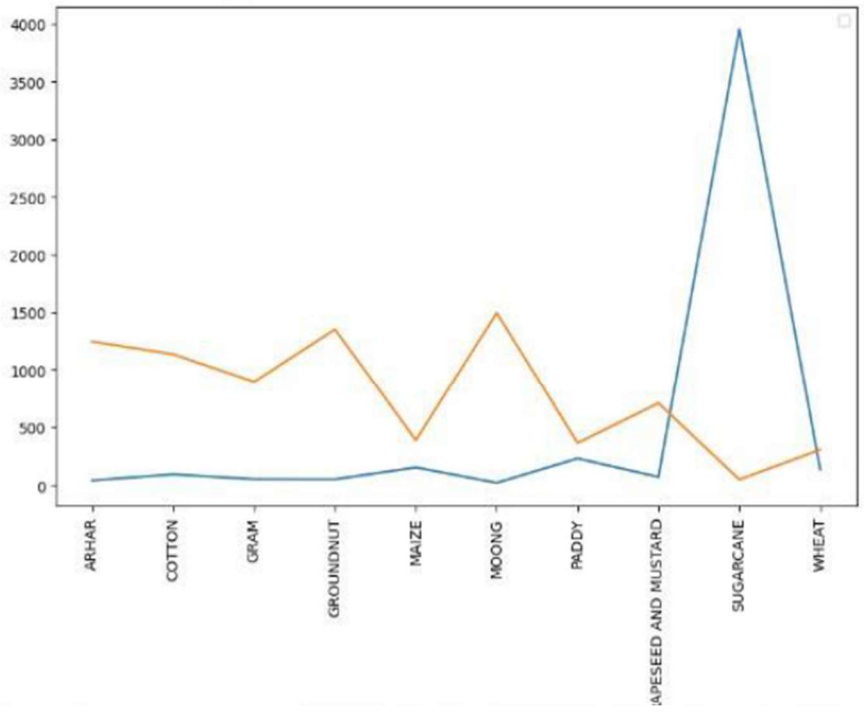
Best Yield Capacity Crop

```
20
21 # Create the layout
22 layout = go.Layout(
23     title='Crop Growth Over Time',
24     xaxis=dict(title='Year'),
25     yaxis=dict(title='Percentage'),
26 )
27
28 # Create the figure
29 fig = go.Figure(data=traces, layout=layout)
30
31 # Show the figure
32 fig.show()
```

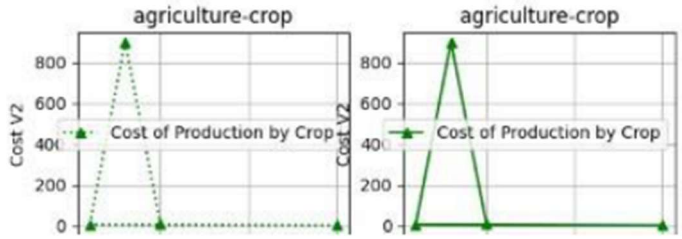

```
fig,axs = plt.subplots(figsize=(10,6))
crop_wise_yield = data.groupby(['Crop']).sum()['Yield (Quintal/ Hectare) ']
plt.plot(crop_wise_yield)
crop_wise_production = data.groupby(['Crop']).sum()['Cost of Production (`/Quintal) C2']/10
plt.plot(crop_wise_production)
plt.xticks(rotation ='vertical')
plt.legend()
```

```
No artists with labels found to put in legend.  Note that artists whose label start with an underscore are ignored when legend() is called with no argument.
```

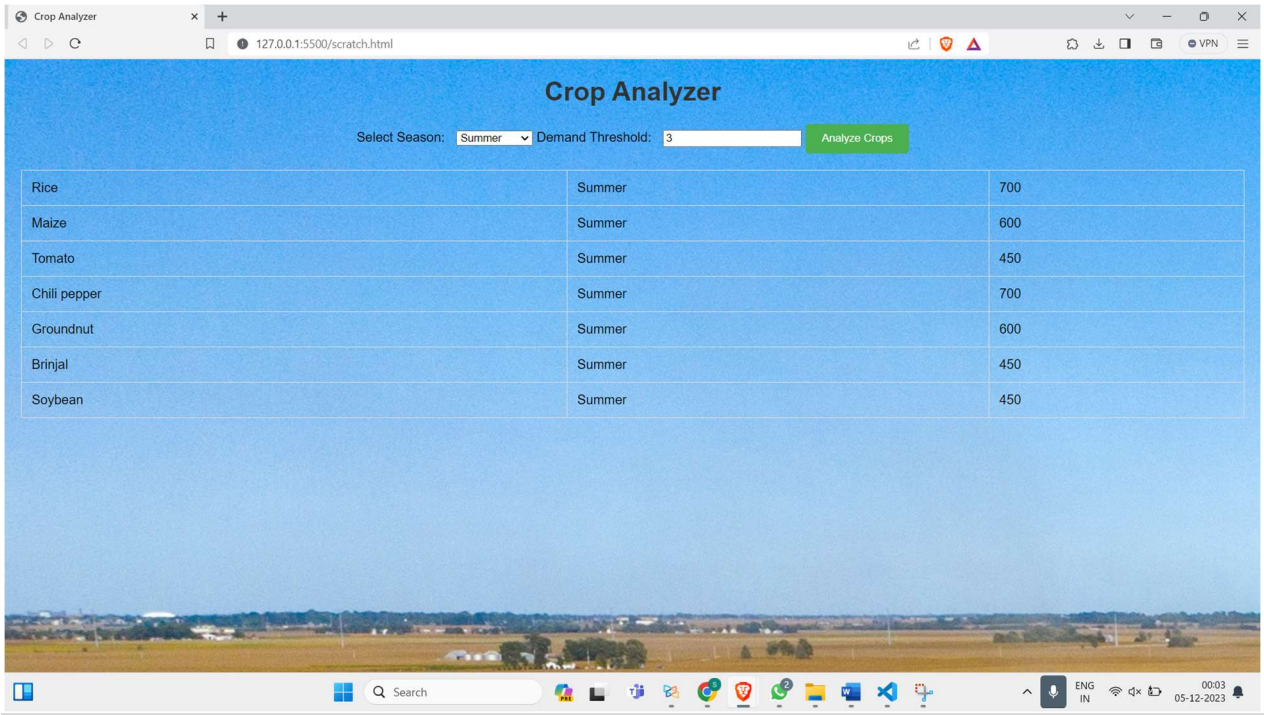
```
<matplotlib.legend.legend at 0x20b507db980>
```



```
import matplotlib.pyplot as plt
x=[20,6,8,10]
y=[1,5,900,11]
# plt.plot(x,y, marker='*', markersize=10, linestyle='dashed', linewidth=1, color='black')
plt.subplot(2,2,1)
plt.plot(x,y, '^:g')
plt.xlabel('Cost ')
plt.ylabel('Cost V2')
plt.title('agriculture-crop')
plt.legend(['Cost of Production by Crop', 'B'])
plt.grid()

plt.subplot(2,2,2)
plt.plot(x,y, '^-g')
plt.xlabel('Cost ')
plt.ylabel('Cost V2')
plt.title('agriculture-crop')
plt.legend(['Cost of Production by Crop', 'B'])
plt.grid()

plt.subplot(2,2,3)
plt.bar(x, y, width=0.2, color='r')
plt.xlabel('Cost ')
plt.ylabel('Cost V2')
plt.title('agriculture-crop')
plt.legend(['Cost of Production by Crop', 'B'])
plt.grid()


plt.subplot(2,2,4)


plt.show()
```



FRONT END:



8. CONCLUTION

The package was designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project:

- Automation of the entire system improves the efficiency
- It provides a friendly graphical user interface which proves to be better when compared to the existing system.
- It gives appropriate access to the authorized users depending on their permissions.
- It effectively overcomes the delay in communications.
- Updating of information becomes so easier
- System security, data security and reliability are the striking features.
- The System has adequate scope for modification in future if it is necessary.

10.REFERENCE

1.Pravar Jain “Artificial Intelligence in Agriculture: Using Modern Day AI to Solve Traditional Farming Problems” in Data Science Blogathon-November 2020.

2.Indian Crops https://www.fao.org/india/fao-in-india/india-at-a-glance/en/

3. Agricultural market (2021) http://agmarknet.gov.in/PriceTrends/Default.aspx. Accessed

4. Production Of Important Crops in Three Largest Producing State https://www.indiabudget.gov.in/economicsurvey/doc/stat/

5. Production Of Important Crops in Three Largest Producing States https://www.safalta.com/blog/major-crops-and-their-states-in-india

6. Major Cropping Seasons https://byjus.com/free-ias-prep/major-croppingseasonsin-india/

7. Major Crops in India https://unacademy.com/content/upsc/studymaterial/indiangeography/major-crops-in-india/

8. Top 10 Agriculture States https://www.tractorjunction.com/blog/top10agriculture-states-in-india/